THIS OPINION WAS NOT WRITTEN FOR PUBLICATION

The opinion in support of the decision being entered today (1) was not written for publication in a law journal and (2) is not binding precedent of the Board.

Paper No. 27

UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

_____

Ex parte NAZEEMUDEEN NOORDEEN and JASON ZHENG

_____

Appeal No. 96-1621
Application No. 08/194,899[1]

_____

ON BRIEF

_____

Before KRASS, JERRY SMITH and TORCZON, Administrative Patent Judges.

KRASS, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on appeal from the final rejection of claims 19, 20, 22 through 25 and 27 through 42, all of the claims pending in the application.

---

[1] Application for patent filed February 14, 1994.

The invention pertains to a precoding and steering mechanism for instructions in a superscaler processor.

Representative independent claim 19 is reproduced as follows:

19. In a computing system, a method comprising the steps of:

(a) fetching a first instruction from a main memory;

(b) fetching a second instruction and a third instruction from main memory;

(c) predecoding the first, second and third instructions to generate predecode bits, wherein the predecode bits include bundling information which indicates whether execution of the second instruction is to be bundled with execution of the first instruction and which indicates whether the execution of the second instruction is to be bundled with execution of the third instruction and wherein the predecode bits additionally include steering information which is in addition to the bundling information, the steering information being used in order to steer each of the first, second and third instructions to one of a first integer arithmetic logic unit, a second integer arithmetic logic unit and a floating point unit for execution; and,

(d) storing the second and third instructions as a double word in an instruction cache, the predecode bits being stored along with the double word in the instruction cache.

The examiner relies on the following references:

Blaner et al. (Blaner) 5,214,763 May 25, 1993

Minagawa et al. (Minagawa), "Pre-Decoding Mechanism For Superscalar Architecture," IEEE Pacific Rim Conference on Communications, Computer and Signal Processing, Vol. 1, Canada (May 9-10, 1991) pp. 21-24.

Claims 19, 20, 22 through 25 and 27 through 42 stand rejected under 35 U.S.C. 103 as unpatentable over Blaner in view of Minagawa.

Reference is made to the many briefs and answers for the respective positions of appellants and the examiner.[2]

OPINION

[2] Previous rejections of the claims under 35 U.S.C. 103 based on Blaner and Hotta and Blaner and Johnson have been withdrawn by the examiner in subsequent answers, the second supplemental answer of May 14, 1996 (Paper No. 19) first indicating the present and sole rejection remaining in the application for our consideration on appeal. Based on the new ground of rejection in the second supplemental answer, appellants amended some of the claims in the third reply brief of June 17, 1996 (Paper No. 20), the amended claims 19, 25, 29, 30, 33, 35 and 37, together with claims 20, 22 through 24, 27, 28, 31, 32, 34, 36 and 38 through 42, as they appear in the appendix to the principal brief, being the claims now on appeal. Thus, the second and third supplemental answers, together with the third and fourth reply briefs, contain the issues and the arguments most relevant to this appeal.

At the outset, we note that claims 31 through 34 improperly depend from a cancelled claim 26. We will leave it to appellants and the examiner to amend the claims for correct dependency at such time as this application may be ready for issue. However, for our purposes, we will presume that claims 31 and 32 depend from independent claim 25 since that appears to be what was intended by the amendment of February 21, 1995 (Paper No. 6) wherein claim 26 was canceled and claims 27 and 29 were made dependent on claim 25.

After a thorough review of the record including, inter alia, the examiner's reasoning and appellants' arguments thereagainst, we will sustain the rejection of claims 19, 20, 22 through 24 and 35 through 42 under 35 U.S.C. 103 but we will not sustain the rejection of claims 25 and 27 through 34 under 35 U.S.C. 103.

With regard to the first group of claims 19, 20 and 23, appellants argue that neither Blaner nor Minagawa discloses the predecoding of three instructions to generate a set of predecode bits which is stored with only two of the instructions. While we recognize the differences between the instant disclosed invention and that taught by Blaner in that

in the former, predecode information for three instructions is stored with *only* two of the instructions in a double word, it is our view that appellants' argument is not commensurate in scope with the instant invention, *as claimed*. The instant claims do not specify that the predecode information for three instructions is stored with *only* two of the instructions in a double word.

Accordingly, predecode information for three instructions may be stored with all three instructions and still be considered to be stored in two of the instructions. Reference to Figure 3 and the corresponding disclosure of Blaner appears to indicate that three consecutive instructions are processed by a compound analyzer, some instructions overlapping between compound analyzers, and that each instruction is given a one-bit tag to determine whether that instruction may be bundled with another instruction. The tag may be considered, broadly, as predecode information and while each instruction in Blaner is given a tag, or predecode information, rather than the predecode information being stored with only two out of three instructions, as claimed, we do not view the claims as

precluding storing the predecode information with all three instructions, which certainly includes two instructions. Since Blaner also discloses storing the instructions in pairs (e.g., column 5, line 45), it is clear that two instructions are stored as a "double word," as claimed. Note, again, contrary to appellants' apparent intention, the claims do not require providing predecode information for three instructions while generating a set of predecode bits which is stored with *only* two of the instructions.

We also do not agree with appellants that Blaner does not use predecode bits to steer instructions. The penultimate sentence of Blaner's abstract appears to indicate that this is precisely what Blaner is doing:

> At instruction issue time, the tag fields of the instructions are examined and those tagged for parallel processing are sent to different ones of the functional units in accordance with the codings of their operation code fields.

In accordance with appellants' grouping of the claims, claims 20 and 23 fall with claim 19.

Turning to claim 22, this claim sets forth six generated predecode bits, each bit indicating bundling of different

instructions and the steering of other instructions to first or second ALUs. We agree with the examiner that it would have been within the skill of the artisan to choose different numbers of predecoded bits to assign to each steering instruction as appellants have not shown that any particular number, viz., six, has any particular advantage over any other number.

In the fourth reply brief, appellants take issue with the examiner's contention, arguing that "the predecode bits, stored with only two instructions, indicate bundling and steering information for three instructions" [fourth reply brief - page 5]. We understand appellants' argument and we can agree that this appears to be a distinction over what is disclosed by the applied references. However, as discussed supra, with regard to claim 19, the claims do not recite language as limiting as appellants' argument would indicate. The claims do not recite that the predecode bits are stored with *only* two instructions while indicating bundling and steering information for three instructions. Moreover, with regard to claim 22, appellants' argument is not relevant to the claim language or to the examiner's rejection. This claim

is directed to the generation of six predecode bits and what those bits indicate. The examiner takes the position that "the assignment of any combination of six predecode bits (or any number of predecode bits) to each steering instruction information would have been an obvious design choice..." The examiner's position appears, to us, to be reasonable and appellants' response does not adequately address the advantages of using six predecode bits and why this particular number is more than a mere design choice.

Accordingly, we will sustain the rejection of claim 22 under 35 U.S.C. 103.

With regard to claim 24, appellants argue [principal brief - page 7] that Blaner does not use predecode bits. However, as set forth supra, it is our view that Blaner does, indeed, disclose such bits for indicating bundling instructions and steering instructions. Thus, we will also sustain the rejection of claim 24 under 35 U.S.C. 103.

We now turn to claim 35 (with which, according to appellants' grouping, claims 38 through 40 and 42 will stand or fall).

Similar to the argument presented with regard to claim 19, appellants contend that neither Minagawa nor Blaner discloses the predecoding of three instructions to generate a set of predecode bits which is stored with *only* two instructions. While we would agree with the argument as setting forth the distinguishing feature of the instant *disclosed* invention over what is taught by the applied references, again, we do not find the instant claim language to be so limiting. The storage of the generated predecode bits with *only* two instructions is not required by claim 35.

Accordingly, we will sustain the rejection of claims 35, 38 through 40 and 42 under 35 U.S.C. 103.

With regard to claim 36 and 37, appellants argue that neither Blaner nor Minagawa discloses that any predecode bits indicate whether two consecutive instructions which are to be bundled for execution are non-aligned or aligned. We disagree.

We turn to page 9 of the instant specification for definitions of “non-aligned” and “aligned”:

> When aligned instructions are bundled, this means that the instruction in the even word of the current

> double word is to be executed simultaneously with the instruction in the odd word of the current double word. When non-aligned instructions are bundled, this means that the instruction in the even word of the current double word is to be executed simultaneously with the instruction in the odd word of the previous double word.

It appears that Blaner describes exactly this at column 5, lines 35-44. Two instructions at a time are processed in parallel:

> A tag bit value of "one" means that the instruction is a "first" instruction. A tag bit value of "zero" means that the instruction is "second" instruction and may be executed in parallel with the proceeding [sic, preceding] first instruction. An instruction having a tag bit value of one may be executed either by itself or at the same time and in parallel with the next instruction, depending on the tag bit value for such next instruction.

Thus, a tag bit value of "zero" might be considered an indication of non-alignment, causing the instruction to be bundled with the preceding instruction while a tag value of "one" might be considered an indication of alignment, wherein the instruction is to be bundled with the next instruction.

Accordingly, we will sustain the rejection of claims 36 and 37 under 35 U.S.C. 103.

Turning now to claim 41, this claim (and arguments for and against) is similar to claim 22. Thus, for the reasons, supra, with regard to claim 22, we will also sustain the rejection of claim 41 under 35 U.S.C. 103.

Finally, we turn to the rejection of claims 25 and 27 through 34 under 35 U.S.C. 103.

Independent claim 25 recites, inter alia, that "the first instruction and the following third instruction are stored as a double word in the instruction cache...." While Blaner appears to disclose the storage of adjacent instructions (an instruction either with its preceding or following instruction) as a double word, we find nothing in Blaner or Minagawa which would disclose or suggest the storage of non-sequential instructions, as the first and third instructions claimed, as a double word in an instruction cache.

Accordingly, we will not sustain the rejection of independent claim 25, or of the claims depending therefrom (27-34) under 35 U.S.C. 103.

CONCLUSION

We have sustained the rejection of claims 19, 20, 22 through 24 and 35 through 42 under 35 U.S.C. 103 but we have not sustained the rejection of claims 25 and 27 through 34 under 35 U.S.C. 103.

The examiner's decision is affirmed-in-part.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 CFR § 1.136(a).

AFFIRMED-IN-PART

ERROL A. KRASS Administrative Patent Judge	)
JERRY SMITH Administrative Patent Judge	) BOARD OF PATENT APPEALS AND INTERFERENCES
RICHARD TORCZON	)

Administrative Patent Judge)

bae

Records Manager
Legal Department, 20BO
Hewlett-Packard Company
P.O. Box 10301
Palo Alto, CA 94303-0890